

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**



KOREAN PATENT ABSTRACTS(KR)

Document Code:A

(11) Publication No.1020020045328

(43) Publication.Date. 20020619

(21) Application No.1020000074751

(22) Application Date. 20001208

(51) IPC Code:

G06F 17/30

(71) Applicant:

INHA UNIVERSITY

(72) Inventor:

PARK, SANG U

YOO, SANG BONG

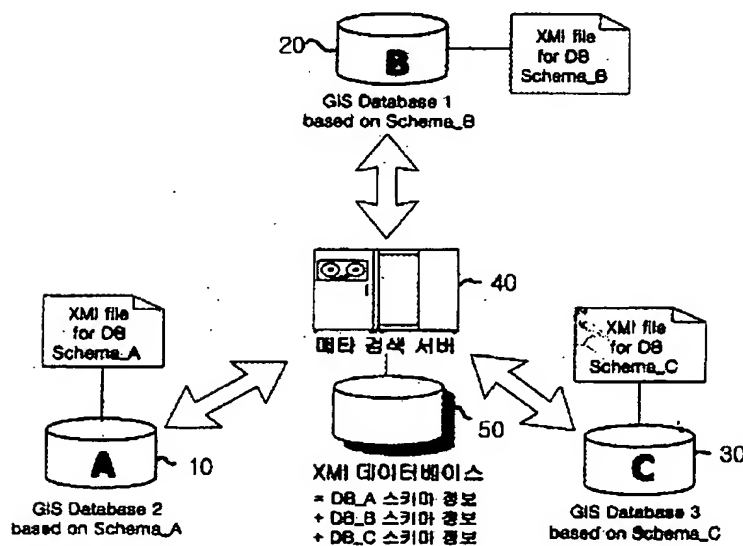
(30) Priority:

(54) Title of Invention

DISTRIBUTED DIFFERENT DATABASE SEARCH METHOD USING XMI

Representative drawing

(57) Abstract:



PURPOSE: A distributed different database search method using an XMI(XML Metadata Interchange) is provided to search a position of a GIS(Geographic Information System) DB and a schema relating to the position.

CONSTITUTION: Each GIS DB (10,20,30) encodes the information into an XML (extensible Markup Language) document of an XMI format by using an XMI conversion program after expressing an independent DB schema as a UML(Unified Modeling Language). IF a URL(Uniform Resource Locator) of each XML file is registered to a meta search server(40), the server stores each DB schema information extracted from the XMI file by an XML

parser in an XMI DB(50). If the user connecting to the server requests a GIS information search, the server dynamically generates an SQL(Structured Query Language) query matching to each DB schema by inquiring to the XMI DB and displays a result of the query to the user.

© KIPO 2003

if display of image is failed, press (F5)

(19) 대한민국특허청(KR) (12) 공개특허공보(A)

(51) Int. Cl. ⁷ G06F 17/30	(11) 공개번호 (43) 공개일자	특2002-0045328 2002년06월 19일
(21) 출원번호	10-2000-0074751	
(22) 출원일자	2000년12월08일	
(71) 출원인	학교법인 인하학원	
(72) 발명자	인천 남구 용현3동 253 인하대학교 유상봉 인천광역시남구용현동253인하대학교자동화공학과정보시스템연구실 박상우 서울특별시서초구방배3동987-1롯데빌라트1차502호	
(74) 대리인	이원희	

심사청구 : 있음

(54) 메타데이터 교환표준을 이용한 분산 이종 데이터베이스검색방법

요약

본 발명은 데이터베이스 검색방법에 관한 것으로, 특히 메타데이터 교환표준(XMI)을 이용한 분산/이종 GIS 데이터베이스 검색방법에 관한 것으로, 다수의 GIS 데이터베이스들과 클라이언트 및 메타 검색 서버가 통신망을 통해 접속되는 메타데이터 검색 시스템에 있어서, 각 GIS 데이터베이스측은 UML로 독자적인 데이터베이스 스키마를 표현한후 XMI 변환 프로그램을 이용하여 XMI 포맷의 XML 문서로 정보를 인코딩하여 저장하는 과정과, 각 XML 파일의 URL을 상기 메타 검색 서버에 등록하면 상기 서버는 XML 파서로 XMI 파일에 내포되어 있는 각 데이터베이스 스키마 정보들을 추출하여 XMI 데이터베이스에 저장하는 과정과, 서버 접속된 임의의 클라이언트로부터 GIS 정보 검색 요청이 있으면, 상기 서버는 상기 XMI 데이터베이스를 조회하여 어떤 GIS 데이터베이스들에게 질의를 보낼 것인지와 각 데이터베이스 스키마에 합당한 SQL 질의문을 동적으로 생성하고, 해당 질의에 따른 결과를 해당 클라이언트에게 전송하여 주는 과정으로 이루어짐을 특징으로 한다.

대표도

도2

색인어

메타데이터, 지리정보, 검색.

명세서

도면의 간단한 설명

- 도 1은 본 발명의 실시예에 따른 메타데이터 검색 시스템 개요도.
- 도 2는 본 발명의 실시예에 따른 메타데이터 검색 시스템 구성도.
- 도 3은 본 발명의 실시예에 따른 메타 검색 시스템 구성에서 클라이언트(60)와 메타 검색 서버(40)의 주요 구성 모듈 예시도.
- 도 4는 본 발명의 실시예에 따른 GIS 메타데이터 표준의 UML 표현 예시도.
- 도 5는 본 발명의 실시예에 따른 SQL 서버 GIS 데이터베이스 스키마의 UML 표현 예시도.
- 도 6은 본 발명의 실시예에 따른 오라클 GIS 데이터베이스 스키마의 UML 표현 예시도.
- 도 7은 본 발명의 실시예에 따른 MiniSQL GIS 데이터베이스 스키마의 UML 표현 예시도.
- 도 8은 본 발명의 실시예에 따른 XMI 데이터베이스의 개념 스키마 예시도.
- 도 9는 본 발명의 실시예에 따른 사용자 인터페이스 예시도
- 도 10은 통합 정보 검색을 위한 사용자 인터페이스 예시도.
- 도 11은 검색결과 가시화를 위한 사용자 인터페이스 예시도.

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 데이터베이스 검색방법에 관한 것으로, 특히 메타데이터 교환표준(XMI)을 이용한 분산/이종 GIS 데이터베이스 검색방법에 관한 것이다.

최근 네트워크의 보급과 지리정보의 활용이 증가하면서 효과적인 지리정보의 교환과 접근에 대한 중요성이 강조되고 있으며, 이를 위해서 GIS(Geographic Information System) 데이터베이스에 지리정보를 저장할 경우에는 해당 GIS 데이터에 대한 메타데이터 정보도 포함시켜 저장하게 된다. 이러한 메타데이터의 사용은 사용자가 원하는 특정 지리정보의 검색을 용이하게 하는 잇점 뿐만 아니라, 방대하고 다양한 지리정보의 체계적인 관리 측면에서도 도움을 주기 때문이다. 하지만 GIS 데이터는 지리적으로 분리된 여러 부서나 조직에서 생성되고 독자적인 데이터베이스 스키마에 의거한 GIS 데이터베이스 상에서 저장 및 관리되고 있기 때문에 정보의 검색기능이 특정 GIS 저장 환경에 종속적이며 정보의 상호운용이 어렵다. 이에 대해 ISO/TC211에서 GIS 메타데이터의 국제 표준이 제정되었으나, 현실적으로는 각기 유사하면서도 자국의 실정에 맞도록 조금씩 차이가 있는 국부적인 GIS 메타데이터 스키마가 사용되고 있는 실정이다.

GIS 메타데이터를 사용하는 대표적인 정보 시스템의 예로는 클리어링하우스로 불리는 공간정보 유통기구가 있는데, 이는 큰 조직이나 국가 수준의 GIS 정보들을 통한 관리하는 시스템으로서, 등록되는 GIS 데이터에 대한 메타데이터를 저장하고 사용자가 특정 정보를 요청할때에 자체 데이터베이스로부터 조건에 일치하는 정보를 검색해 준다.

그러나 상술한 클리어링하우스도 국가나 조직간에 차이가 있는 메타데이터 스키마를 사용하기 때문에 동일한 GIS 메타데이터를 사용하는 국가나 조직에 대해서만 검색 서비스가 지원되는 한계가 있다. 한편, 사용자 입장에서는 찾고자 하는 정보가 지리적으로 어느 GIS 데이터베이스에 있는지와, 그 데이터베이스에 저장되어 있는 GIS 정보가 어떠한 스키마에 의해서 저장되었는가를 모르더라도 자신이 원하는 정보를 질의하고 그 결과를 얻을 수 있어야만 한다.

발명이 이루고자하는 기술적 과제

따라서 본 발명의 목적은 상술한 문제점을 해결하기 위하여 안출된 것으로서, 특정 GIS 데이터베이스의 위치와 그에 관련되는 스키마에 대해 투명한 검색기능을 제공할 수 있는 메타데이터 교환표준을 이용한 분산 이종 데이터베이스 검색방법을 제공함에 있다.

본 발명의 또 다른 목적은 사용자에게 특정 GIS 데이터베이스의 위치 뿐만 아니라 메타데이터 스키마에 투명한 포괄적 검색 기능을 제공함은 물론, 특정 GIS 데이터베이스 스키마에 종속적인 정보까지도 검색할 수 있는 메타데이터 교환표준을 이용한 분산 이종 데이터베이스 검색방법을 제공함에 있다.

발명의 구성 및 작용

상기 목적을 달성하기 위한 본 발명은 다수의 GIS 데이터베이스들과 클라이언트 및 메타 검색 서버가 통신망을 통해 접속되는 메타데이터 검색 시스템에 있어서,

각 GIS 데이터베이스측은 UML로 독자적인 데이터베이스 스키마를 표현한후 XMI 변환 프로그램을 이용하여 XMI 포맷의 XML 문서로 정보를 인코딩하여 저장하는 과정과,

각 XML 파일의 URL을 상기 메타 검색 서버에 등록하면 상기 서버는 XML 파서로 XMI 파일에 내포되어 있는 각 데이터베이스 스키마 정보들을 추출하여 XMI 데이터베이스에 저장하는 과정과,

서버 접속된 임의의 클라이언트로부터 GIS 정보 검색 요청이 있으면, 상기 서버는 상기 XMI 데이터베이스를 조회하여 어떤 GIS 데이터베이스들에게 질의를 보낼 것인지와 각 데이터베이스 스키마에 합당한 SQL 질의문을 동적으로 생성하고, 해당 질의에 따른 결과를 해당 클라이언트에게 전송하여 주는 과정으로 이루어짐을 특징으로 한다.

이하 본 발명의 바람직한 실시예를 첨부한 도면을 참조하여 상세히 설명하기로 한다. 하기 설명 및 첨부도면에서 GIS 데이터베이스의 수, 사용자 인터페이스의 예 및 구체적인 처리 흐름등과 같은 많은 특정 상세들이 본 발명의 보다 전반적인 이해를 제공하기 위해 나타나 있다. 이들 특정 상세를 없이 본 발명이 실시될 수 있다는 것은 이 기술분야에서 통상의 지식을 가진 자에게 자명할 것이다. 한편 본 발명의 요지를 불필요하게 흐릴 수 있는 공지 기능 및 구성에 대한 상세한 설명은 하기에서 생략하기로 한다.

우선 도 1은 본 발명의 실시예에 따른 메타데이터 검색 시스템의 개요도를 도시한 것으로, 본 발명의 실시예에 따른 메타데이터 교환표준을 이용한 분산 이종 데이터베이스 검색방법을 간략히 설명하기 위해 도시한 도면이다. 도 1에서, 각각의 GIS 데이터베이스측(10, 20, 30)은 UML(Unified Modelling Language)로 독자적인 데이터베이스 스키마를 표현한후, XMI(메타데이터 교환표준) 변환 프로그램을 이용하여 XMI 포맷의 XML 문서로 정보를 인코딩한다. 그리고 각 XML 파일의 URL(Universal Resource Location)을 메타 검색 서버(40)에 등록하면, 상기 서버(40)는 XML 파서(parser)로 XMI 파일에 내포되어 있는 각 데이터베이스 스키마 정보들(클래스와 속성정보)을 추출하여 XMI 데이터베이스(50)에 저장한다. 그리고 만약 상기 서버(40)에 접속한 임의의 사용자가 GIS 정보 검색을 요청하면, 상기 서버(40)는 XMI 데이터베이스(50)를 조회하여 어떤 GIS 데이터베이스들에게 질의를 보낼 것인지와 각 데이터베이스 스키마에 합

당한 SQL(Structured Query Language) 질의문을 동적으로 생성하고 해당 질의에 따른 결과를 사용자에게 보여 준다.

이하 상술한 본 발명의 개요를 좀 더 구체적으로 설명하면 다음과 같다.

도 2는 본 발명의 실시예에 따른 메타데이터 검색 시스템 구성을 도시한 것이다. 도 2를 참조하면, 우선 사용자(60)측은 메타 검색 서버(40)에 접속하게 되면 애플릿(Applet)을 로드하고 CORBA(Common Object Request Broker Architecture) 클라이언트로서 동작을 하게 된다. 미들웨어는 CORBA를 사용했으며, 이는 XMI 파일로부터 특정 GIS 데이터베이스 스키마 정보의 추출, XMI 데이터베이스와의 인터페이싱, 그리고 후단(back-end)에 있는 서로 다른 종류의 GIS 데이터베이스들과 JDBC(Java DataBase Connectivity)를 통한 정보 조회 및 반환을 담당하는 역할 등을 담당한다. XMI 데이터베이스(50)는 각 GIS 데이터베이스 스키마정보와 기타 부수 정보(접속계정, URL 등)들을 통합관리하는 저장장소로서, 구현에서는 객체 저장(Object Store) OODBMS(Object Oriented DataBase Management System)를 사용한다.

한편 구현환경으로서, 클라이언트(60)측은 스윙(swing) 기반의 자바 애플릿이기 때문에 인터넷이 가능한 환경이면 어디든 가능하다. 메타 검색 서버(40)측은 Solaris 2.6하에서 OrbixWeb 3.0을 CORBA로 사용하였고, XMI 데이터베이스(50)로는 Object Store 5.1 OODBMS를 사용하였다. 기타 관계형 데이터베이스와의 JDBC 접속을 위한 JDBC 드라이버들은 네가지 타입중 유형 3,4에 속하는 드라이버들을 선택사용하였다. 그리고 GIS 데이터베이스(50)로는 SQL 서버 7, Oracle 7.3, MiniSQL 1.0 등 세 개의 관계형 데이터베이스만을 사용하였다. 자바는 JDK 1.1.7이고, XML 파서는 오라클 파서 2.0, 그리고 각 데이터베이스의 개념 스키마 모델링에는 Rational Rose UML 툴과 XMI 인코딩에는 IBM에서 제공하는 XMI 변환 툴킷을 사용하였다.

이하 클라이언트(60)와 메타 검색 서버(40)의 주요 구성 모듈을 도 3을 참조하여 설명하기로 한다.

도 3은 본 발명의 실시예에 따른 메타 검색 시스템 구성에서 클라이언트(60)와 메타 검색 서버(40)의 주요 구성 모듈 예시도를 나타낸 것으로, 보다 상세하게는 클라이언트(60)측과 서버(40)측에서 처리해야 하는 작업을 수행하기 위한 기능 모듈들을 도시한 것이다.

도 3을 참조하면, 우선 클라이언트(60)측의 새로운 GIS 데이터베이스 등록처리 모듈(62)은 등록자가 특정 GIS 데이터베이스를 검색하고 서버(40)에 등록하도록 하기 위한 인터페이스와 정보 처리 기능을 제공한다. 분류 검색 및 데이터베이스별 검색을 위한 UI의 동적생성모듈(64)은 XMI 데이터베이스(50)에 있는 정보 조회를 통해 사용자에게 제공되는 다양한 분류 검색 인터페이스와 관련 결과 프레임을 생성하여 주는 역할을 수행한다. CORBA 통신을 위한 입출력 데이터 처리모듈(66, 46)은 CORBA 시스템 양단간에 통신하는 데이터나 메시지의 처리를 담당한다.

한편 메타 검색 서버(40)측의 XMI 파일로부터 GIS 메타 데이터 스키마 정보를 추출하는 모듈(42)은 새로 이 등록되는 GIS 데이터베이스의 XMI 파일로부터 자동으로 스키마 정보(구현에서는 클래스와 관련 속성 정보)를 추출하여 데이터베이스화 한다. 분류검색을 위한 자동 SQL 생성 모듈(44)은 사용자가 입력한 키워드와 검색 옵션에 따라 질의를 보낼 특정 GIS 데이터베이스마다 질의문을 생성한다. JDBC를 통한 질의 처리 모듈(45)은 다양한 종류의 관계형 GIS 데이터베이스에 대한 접속/질의/결과의 반환을 담당한다.

이하 본 발명의 적용예를 설명하면, 우선 도 4는 본 발명의 실시예에 따른 GIS 메타데이터 표준의 UML 표현을 예시한 것으로, 보다 상세하게는 미국 FGDC의 CSDGM(Content Standards for Digital Geospatial Metadata)와 한국 국립 지리원의 외부 메타데이터를 참조하여 추출한 주요 메타데이터 섹션들과 수치데이터와의 관계를 UML로 표현한 것이다. 중앙의 디지털데이터세트(DigitalDataSet)를 수치데이터에 대한 참조 정보로서 간주한다면, 그 주변에 <<include>>라는 스테레오타입으로 표현된 7개의 클래스는 앞서 언급했던 메타데이터 표준들로부터 얻은 섹션들과 각 섹션에 포함되는 부속 정보들이다. 디지털데이터세트 클래스의 속성 중 mapID와 typeOfMapID는 각각 수치데이터를 구분할 수 있는 mapIDm1 유형과 식별자를 의미한다. 예로서, mapID는 GIS 데이터베이스에 대한 경우 메타데이터와 관련 있는 수치데이터 테이블에 대한 외래키 또는 객체 참조가 될 수 있고, 단순한 GIS 메타데이터베이스라고 한다면 도면번호와 같이 임의의 수치데이터를 구별할 수 있는 식별자 역할을 한다. 하나의 디지털데이터세트, 즉 수치데이터에 관련도니 메타데이터 섹션들은 자료식별정보(Identification), 데이터품질(DataQuality), 연혁정보(Lineage), 공간참조정보(SpatialReferenceInformation), 메타데이터참조정보(MetadataReference), 공간데이터구성정보(SpatialDataOrganization), 배포정보(DistributionInformation) 등으로 분류하였으며, 각 메타데이터 클래스의 속성들에 대한 데이터 타입은 문자열로 설정하였다.

도 5는 본 발명의 실시예에 따른 SQL 서버 GIS 데이터베이스 스키마의 UML 표현 예시도로서, 보다 상세하게는 SQL 서버 데이터베이스에 대한 수치데이터와 관련 메타데이터 스키마의 UML 표현이다. 본 발명에서는 관계형 데이터베이스를 대상으로 하였기 때문에 테이블마다 기본키가 있으며, 키값의 타입은 int(정수형)로 설정했고 키의 이름은 디지털데이터세트를 제외한 모든 테이블에 대해서 테이블의 이름 + ID로 구성하였다. 기타 부수 속성들의 타입은 단순화를 위해서 모두 문자열로 설정하였다. 물론 실제로 데이터베이스에 GIS 정보를 저장할 경우에 사용하는 데이터 타입은 각 DBMS가 제공하는 적절한 수치나 문자열의 타입을 선택하여 사용하게 된다.

도 6은 본 발명의 실시예에 따른 오라클 GIS 데이터베이스 스키마의 UML 표현이며, 도 7은 MiniSQL GIS 데이터베이스 스키마의 UML 표현 예시도이다.

앞서 설명한 세 개의 데이터베이스 스키마들의 차이를 간단히 비교해 보면, SQL 서버의 스키마와 오라클의 스키마에는 식별자(Identification) 클래스가 공통으로 있지만, 각 데이터베이스 스키마는 식별자를 제외한 비공통 메타데이터 클래스를 한 개씩 갖도록 구성을 했다. MiniSQL의 스키마는 식별자 클래스가 있지만, SQL 서버나 오라클의 스키마에서 정의한 식별자 클래스와 속성의 개수에서 차이가 있다. 그리고 SpatialReferenceInfo 클래스는 앞서 언급한 두 개의 다른 데이터베이스 스키마에 없는 클래스이다.

일단, 각 메타데이터 스키마 부분을 UML로 표현한 후에 IBM에서 제공하는 XMI 툴킷을 사용하여 UML 메타

모델에 의거한 XMI 파일로 변환을 하게 된다.

이하 XMI 파일로부터 스키마 정보를 추출하는 과정을 설명하면, 우선 XMI 표준 포맷은 기본적으로 적격 이면서(well-formed) 유효한 XML 문서이며, 일반적으로 다음과 같은 구조를 갖는다.

먼저 DTD(Document Type Definition)의 경우 본 시스템의 각 GIS 데이터베이스 스키마로부터 변환된 XMI파일은 UML에 의거하기 때문에, 외부 DTD로서 uml.dtd를 참조한다. 그리고 최상위의 루트 요소로서 XMI를 갖는다. 루트 요소 밑에는 여러 가지 모델링 요소들(클래스, 속성, 메소드 등)과 관련 모델링 요소들의 특성(가시성, 관계종류, 도수 등)들이 선택적으로 기술될 수 있도록 되어 있다. 이러한 스키마 정보를 갖는 XMI 파일의 URL을 메타 검색 서버(40)에 등록하면, 서버(40)측은 XML파서를 사용하여 데이터베이스의 메타데이터 스키마 정보를 파싱하여 얻어낸다. XML파서는 크게 DOM(Document Object Model)과 SAX(Simple API for XML)가 존재하는데, 본 시스템의 구현 모듈에서는 SAX의 이벤트 기반 API를 사용하여 GIS 데이터베이스 스키마 정보중 클래스와 관련 속성 정보들을 추출하고, XMI 데이터베이스화하도록 하였다.

XMI 데이터베이스의 설계에 대하여 언급해 보면, 본 발명의 실시예에서는 우선 XMI 데이터베이스로 Object Store OODBMS를 사용하였다. 도 8은 XMI 데이터베이스의 개념 스키마 중 일부 중심 클래스들과 관련 오퍼레이션들을 UML 클래스 다이어그램으로서 표현한 것이다. 도 8에 도시된 모든 클래스가 영속성을 지니기는 하지만 데이터베이스루트(DatabaseRoot)와 스키마루트(SchemaRoot)는 임의의 GIS 데이터베이스에 독립적인 역할을 하고, xxxDescriptor는 특정 GIS 데이터베이스에 종속적인 정보를 관리한다.

데이터베이스루트는 데이터베이스의 세션이 시작된 후, 각 데이터베이스디스크립터의 정보에 접근하기 위한 반복자 역할의 엔트리 객체이다. 스키마루트는 등록된 데이터베이스 스키마가 갖는 속성들의 합집합을 관리하도록 구성하였다. 따라서 새로이 등록되는 데이터베이스 스키마중 임의의 클래스의 속성이 스키마루트가 관리하고 있는 기존의 것과 같은 이름과 타입을 지닌다면, 단지 기존에 생성된 AttrDescriptordp 대한 참조만을 지니게 된다. 이러한 편의 클래스를 사용하게 되면, 사용자 인터페이스의 동적 구성이나 질의 자동 생성시에 기존의 모든 데이터베이스 스키마를 반복적으로 조사해야 하는 연산의 복잡함을 덜 뿐만 아니라, 각 데이터베이스 스키마의 속성 중복으로 야기되는 불필요한 데이터베이스 공간 낭비를 막을 수 있다. UML 표현중에서 연관(association)은 모두 영속적 영역에 저장된 객체에 대한 참조이다.

한편 검색 키워드에 대한 SQL 문장의 자동 생성 메카니즘에 대하여 설명하면, 본 발명에서는 GIS 데이터베이스 스키마들이 하나의 디지털데이터세트라는 수치데이터 참조 테이블을 중심으로 기타 메타데이터 세션들과 각 1:1 관계로 구성되도록 하였다. 질의문을 보낼 특정 데이터베이스 스키마에 대해, 디지털데이터세트를 포함한 총 메타데이터 관련 테이블 수를 n이라 하고, k를 사용자가 입력한 검색 키워드의 개수라고 할때(각 검색 키워드가 스키마 내의 각기 다른 테이블의 속성과 연관된다고 가정), 아래의 텍스트 박스와 같은 형태로 질의문이 생성된다.

sel_table(1), sel_table(k).sel_field와 같이 요소 앞에 sel_이 붙은 항목들은 검색 키워드가 입력된 테이블 또는 속성과 관련이 있는 이름을 나타내며, keyword_1, keyword_k 등은 입력된 키워드에 대한 식별기호이다.

우선 선택된 하나의 테이블과 속성에 대해서 검색을 통해 해당 레코드들을 얻은 후 관계 연산을 통해서 디지털데이터세트 테이블(수치정보 참조 테이블)의 키를 얻는다. 일단 검색결과와 관련된 디지털데이터세트의 기본 키들을 얻게 되면 나머지 테이블들로 디지털데이터세트 테이블을 중심으로 조인 연산을 통해서 연관된 모든 레코드들을 얻게 된다. 따라서 JDBC를 통해 처리된 질의 결과는 모든 관련 메타데이터 테이블들의 조인을 통해 하나의 테이블로서 반환된다. 그리고 결과 데이터는 코바를 통해서 클라이언트 측으로 전송되고, 애플릿은 각 데이터베이스의 결과들을 받아서 결과 프레임에는 각 데이터베이스 사이의 관련 정보와 관련 수치데이터의 메타데이터를 일괄적으로 보여 주게 된다.

따라서 각 관계형 DBMS에 대해서 전송되는 질의요청과 정보 반환은 한번의 JDBC 접속으로서 끝나게 된다.

본 발명의 실시예에서 자동 생성된 SQL문의 형태를 예시하면 아래와 같다.

```
SELECT * (또는 table[1].*, table[2].*,...,table[n].*)
FROM      table[1], table[2],...,table[n-1], table[n]
WHERE      table[1].id={DigitalDataSet}.id
AND        ...
AND        table[n-k-1].id={DigitalDataSet}.id
AND        {DigitalDataSet}.id=sel_table[1].id
AND        [UPPER]sel_table[1].sel_field[LIKE][UPPER]('%keyword_1%')
AND        ...
AND        {DigitalDataSet}.id = sel_table[k].id
AND        [UPPER]sel_table[k].sel_field[LIKE][UPPER]('%keyword_k%')
```

이하 도 9 내지 도 11을 참조하여 본 발명의 실시예에 따른 사용자 인터페이스에 대하여 부연 설명하기로 한다.

우선 도 9는 본 발명의 실시예에 따른 사용자 인터페이스 예를 나타낸 것이며, 도 10은 통합 정보 검색을 위한 사용자 인터페이스 예를 도시한 것이다. 그리고 도 11은 검색결과 가시화를 위한 사용자 인터페

이스 예를 도시한 것이다.

도 9를 참조하여 새로운 GIS 데이터베이스 등록을 설명하면, 우선 등록을 선택하게 되면 미리 준비된 등록 관련 인터페이스가 도 9와 같이 나타나고, 사용자는 입력란에 자신이 등록하려는 데이터베이스의 위치, 종류, 접속을 위한 사용자 계정, 그리고 XMI 파일의 URL을 기입한다. XMI 파일은 등록하려는 GIS 데이터베이스의 스키마중 메타데이터 부분을 UML로 모델링하고 XMI 변환을 한 결과이다. 서버(40)측에서는 입력된 정보를 전송받고, XML파서를 사용하여 XMI파일로부터 클래스와 속성들의 정보를 추출한후, 데이터베이스 정보와 추출된 스키마 정보를 데이터베이스에 저장하게 된다.

도 10을 참조하여 통합 분류 검색 인터페이스에 대하여 설명하면, 사용자는 검색을 원하는 속성 옆의 입력란에 키워드를 입력한후, 상단에 있는 검색 옵션을 설정하고 서치버튼을 선택하면 애플릿에서는 입력란이 널(null)이 아닌 텍스트 필드들에 대해서만 클래스 이름, 속성 이름, 키워드를 메시지화하여 코바 서버측으로 전송하게 된다.

도 11을 참조하여 검색결과 인터페이스에 대하여 설명하면, 클라이언트의 질의에 대해 서버측에서는 전달된 검색어 메시지와 XMI 데이터베이스에 있는 스키마 정보를 조회하여, 질의 요청을 보낼 데이터베이스의 목록과 접속정보를 얻고 구조체 정보를 자동으로 SQL 쿼리를 구성해 주는 모듈로 전달한다. 이때 쿼리는 RDB에서 특정 테이블중 하나의 속성에 대해서 행해지며, JDBC를 통해 처리된 결과는 모든 메타데이터 테이블의 조인으로 이루어진 하나의 결과세트(resultset)으로 반환되도록 하였다. 서버측의 결과가 클라이언트 애플릿으로 전송되면 애플릿은 도 11에 도시한 바와 같이 질의 결과를 얻은 GIS 데이터베이스의 부수정보와 검색 결과를 하나의 테이블로 통합하여 보여 준다.

따라서 사용자는 찾고자 하는 정보가 지리적으로 어느 GIS데이터베이스에 있는지와, 그 데이터베이스에 저장되어 있는 GIS 정보가 어떠한 스키마에 의해서 저장되었는가를 모르더라도 자신이 원하는 정보를 질의하고 그 결과를 얻을 수 있게 되는 것이다.

발명의 효과

상술한 바와 같이 본 발명은 다양한 GIS 데이터베이스의 위치와 그에 관련되는 각 데이터베이스 스키마에 대하여 투명한 검색기능을 제공할 수 있는 이점이 있는데, 이는 GIS 메타데이터와 OMG의 메타데이터 교환 표준인 XMI를 이용하여 임의 데이터베이스의 스키마 정보를 분석하여 메타 검색 서버측에 데이터베이스화 한후, 필요한 경우 이에 대한 조회를 통해서 사용자에게 제공함으로써, 향상된 분류 검색 서비스와 여러 GIS 데이터베이스에 대한 투명성을 제공할 수 있게 되는 것이다.

또한 구현상에서 미들웨어로는 CORBA를 사용하였고 후단(back-end)의 다양한 GIS 데이터베이스와 코바 서버측과의 통신 메커니즘은 JDBC를 이용하기 때문에 메타 검색 서버는 특정 플랫폼에 독립적이며, 높은 확장성을 갖는 이점이 있다. 그리고 클라이언트측 인터페이스로는 플랫폼에 독립적인 자바 애플릿을 사용하기 때문에 사용자들에게 서버로의 접근과 사용이 용이한 웹 기반의 인터페이스를 제공하는 효과도 있다.

(57) 청구의 범위

청구항 1

다수의 GIS 데이터베이스들과 클라이언트 및 메타 검색 서버가 통신망을 통해 접속되는 메타데이터 검색 시스템에 있어서,

각 GIS 데이터베이스측은 UML로 독자적인 데이터베이스 스키마를 표현한후 XMI 변환 프로그램을 이용하여 XMI 포맷의 XML 문서로 정보를 인코딩하여 저장하는 과정과,

각 XML 파일의 URL을 상기 메타 검색 서버에 등록하면 상기 서버는 XML 파서로 XMI 파일에 내포되어 있는 각 데이터베이스 스키마 정보들을 추출하여 XMI 데이터베이스에 저장하는 과정과,

서버 접속된 임의의 클라이언트로부터 GIS 정보 검색 요청이 있으면, 상기 서버는 상기 XMI 데이터베이스를 조회하여 어떤 GIS 데이터베이스들에게 질의를 보낼 것인지와 각 데이터베이스 스키마에 합당한 SQL 질의문을 동적으로 생성하고, 해당 질의에 따른 결과를 해당 클라이언트에게 전송하여 주는 과정으로 이루어짐을 특징으로 하는 메타데이터 교환표준(XMI)을 이용한 분산 이종 데이터베이스 검색방법.

청구항 2

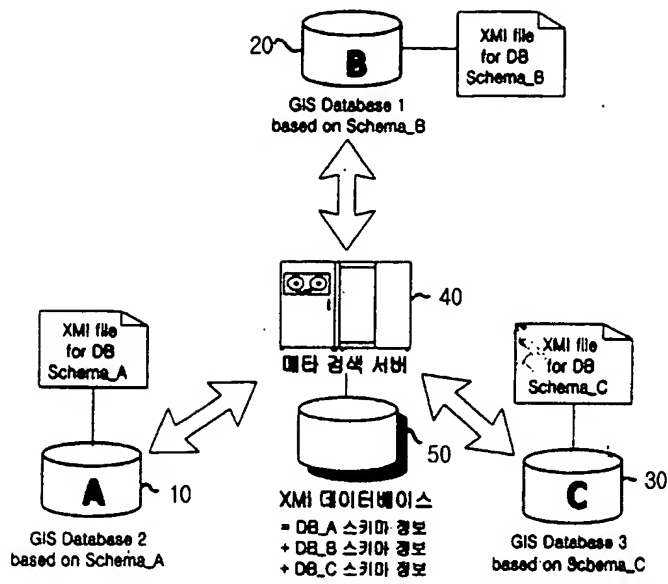
제1항에 있어서, 상기 XMI 데이터베이스는 각 GIS 데이터베이스 스키마정보와 접속 계정, URL과 같은 부수 정보를 통합관리하는 저장장소로서 객체 저장(Object Store) OODBMS(Object Oriented DataBase Management System)로 구현됨을 특징으로 하는 메타데이터 교환표준(XMI)을 이용한 분산 이종 데이터베이스 검색방법.

청구항 3

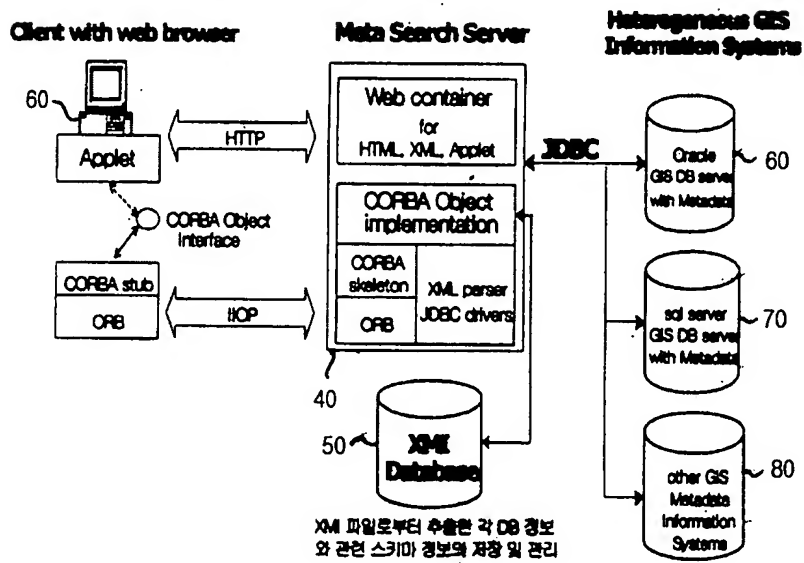
제1항에 있어서, 상기 각각의 데이터베이스 스키마 정보들에는 클래스와 속성정보들이 적어도 포함됨을 특징으로 하는 메타데이터 교환표준을 이용한 분산 이종 데이터베이스 검색방법.

도면

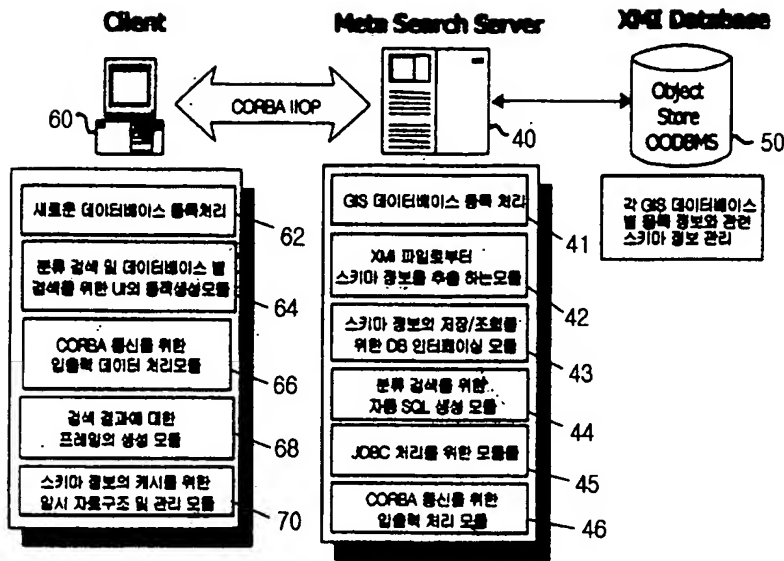
도면 1



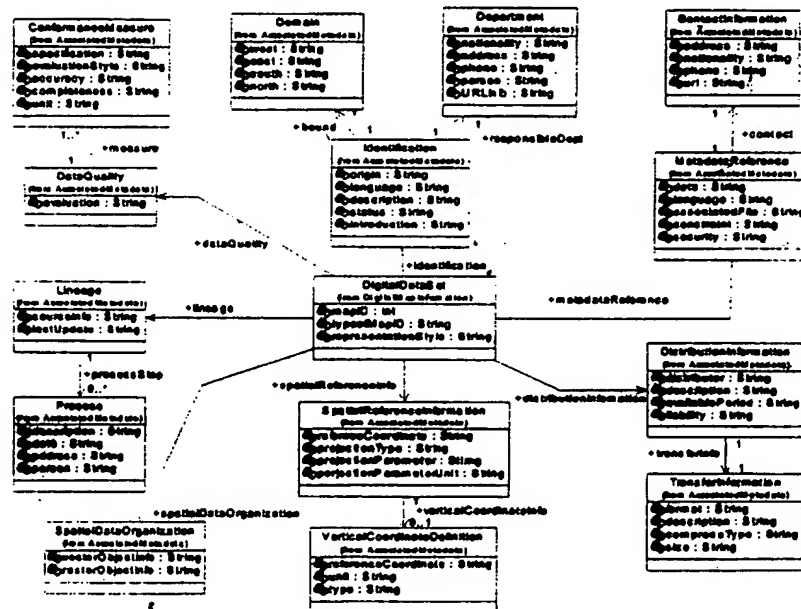
도면2



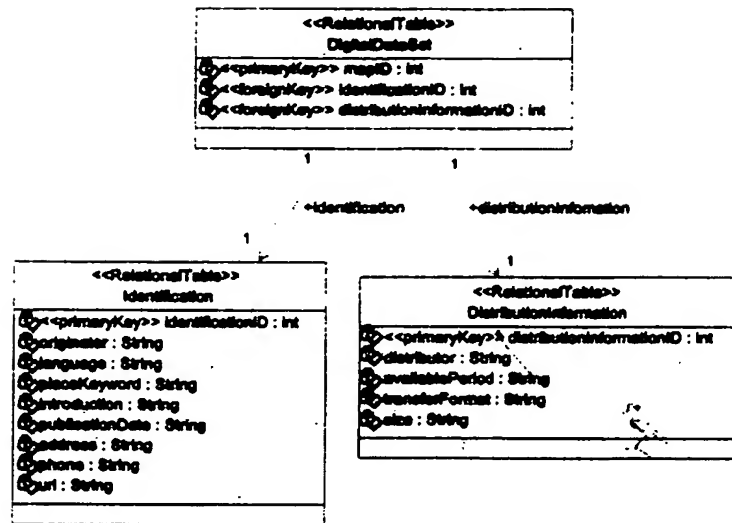
도면3



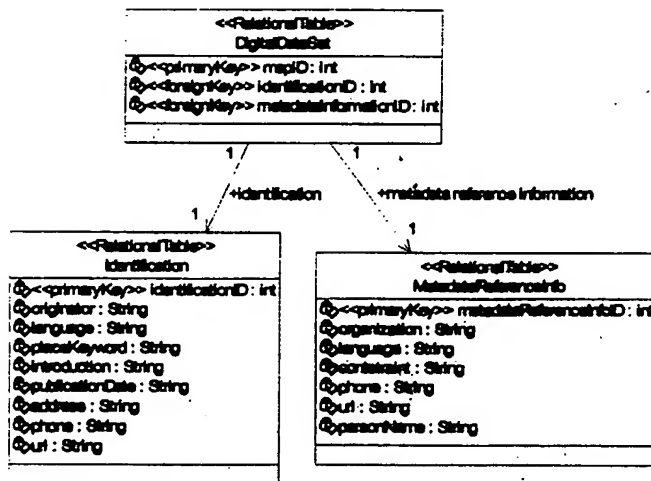
도면4



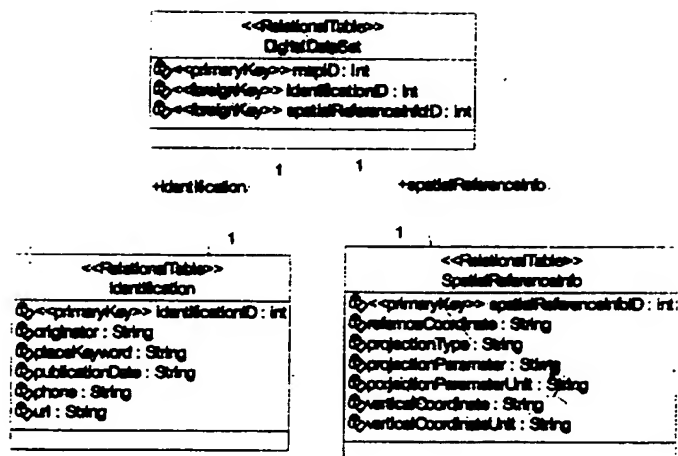
도면5



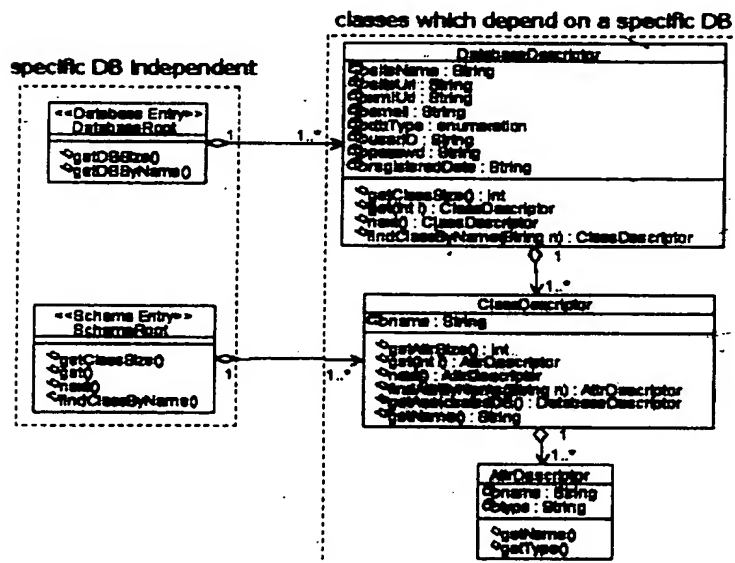
도면6



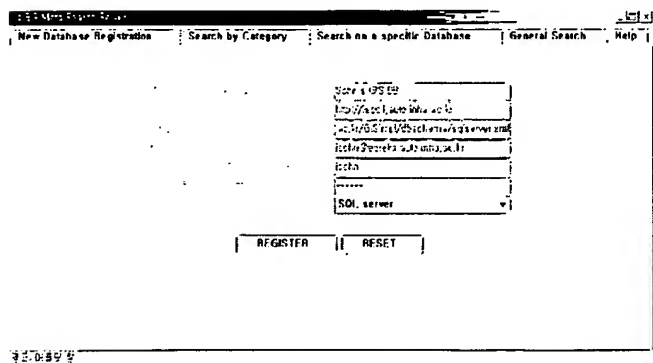
도연7



도면8



도면9



도면 10

The screenshot shows the 'New Database Registration' dialog box in the 'Data' tab. The 'Classification' tree on the left is expanded to 'Ordinal'. The 'Search by Category' tab is selected. The 'Send Query' button is highlighted. The 'Search on a specific Database' tab is also visible. The 'General Search' tab is selected. The 'Help' button is visible. The 'Search' button is visible.

도면Y1

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

311

312

313

314

315

316

317

318

319

320

321

322

323

324

325

326

327

328

329

330

331

332

333

334

335

336

337

338

339

340

341

342

343

344

345

346

347

348

349

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

370

371

372

373

374

375

376

377

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

414

415

416

417

418

419

420

421

422

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437

438

439

440

441

442

443

444

445

446

447

448

449

450

451

452

453

454

455

456

457

458

459

460

461

462

463

464

465

466

467

468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

483

484

485

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

501

502

503

504

505

506

507

508

509

510

511

512

513

514

515

516

517

518

519

520

521

522

523

524

525

526

527

528

529

530

531

532

533

534

535

536

537

538

539

540

541

542

543

544

545

546

547

548

549

550

551

552

553

554

555

556

557

558

559

560

561

562

563

564

565

566

567

568

569

570

571

572

573

574

575

576

577

578

579

580

581

582

583

584

585

586

587

588

589

590

591

592

593

594

595

596

597

598

599

600

601

602

603

604

605

606

607

608

609

610

611

612

<